**National University of Computer & Emerging Sciences**

**Karachi Campus**



**Fake News Detection with Sentiment Analysis**

Project Description

Artificial Intelligence AI2002

Section: BSCS-6A

**Group Members:**

**22K-4557 Taqwa Rasheed**

**22K-4538 Fatimah Ansari**

**Abstract**

There is a lot of news across various social media platforms and one of the biggest challenges in this era is to tell if the news you read is legitimate or fake. This is due to the complexity of natural language and diversity in the sources of information. We tried to address this problem by applying machine learning algorithms combined with sentiment analysis to classify news statements as true, false, or partially true. We used several classification models including Naive Bayes, Decision Tree, Logistic Regression, and K-Nearest Neighbors. There was class imbalance in the dataset which was handled using over-sampling technique SMOTE. The results demonstrated that oversampling improved accuracy slightly and the overall classification performance remained moderate, highlighting the difficulty of detecting the fake news.

**Introduction**

The rise in the usage of social media and online news outlets has revolutionized the spread of information. However, it also facilitated the rapid spread of misinformation. Identifying fake news manually has become a challenge due to the amount of content that is generated daily. As a result, automated approaches like using machine learning techniques to identify the fake news have become crucial. This project explores the application of machine learning models and sentiment analysis to detect fake news from textual statements.

**Background**

Existing papers have employed various techniques for detecting the fake news, ranging from feature engineering to complex deep learning models. Classical machine learning classifiers like Naive Bayes, Support Vector Machines, and Decision Trees have been effective for basic text classification tasks. Some papers also considered sentiment analysis, with the hypothesis that fake news often exhibits extreme emotional tones. Nonetheless, challenges like class imbalance, subtle linguistic patterns, and sarcasm detection remain significant obstacles in fake news detection.  
  
**Methods and Materials**

1. *Dataset*

The dataset that we used is the LIAR dataset, consisting of short political statements labeled into six categories (‘Pants-On-Fire’, ‘False’, ‘Barely-True’, ‘Half-True’, ‘Mostly-True’, ‘True’). It has 10,000+ rows and 13 columns.

1. *Preprocessing*

The Liar dataset reflects the real-world data hence it is very noisy and needs processing and cleaning. We labeled all columns after understanding each feature, selected only the ‘Statement’ and ‘Label’ columns, checked for missing values and converted the categorical label into numerical data for further processing. Refer to figure 1

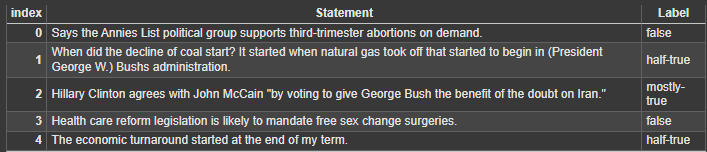


Figure 1

1. *Feature Extraction*

We used Term Frequency-Inverse Document Frequency (TF-IDF) to convert the textual data into numerical format to apply machine learning models on it.

1. Sentiment Analysis

We added a feature of sentiment analysis by generating the sentiment score using the ‘TextBlob’ library of python.

1. Models Applied

We applied the following models

* 1. Naive Bayes
  2. Decision Tree
  3. K-Nearest Neighbors (neighbors=5)
  4. Logistic Regression

We also explored KMeans clustering to identify the groupings of the statements

1. *Data Balancing*

We checked for class distribution of the ‘Label’ feature and found it was imbalanced and then used oversampling technique SMOTE for the minority classes to make the data more balanced. Refer to figure 2 and 3.

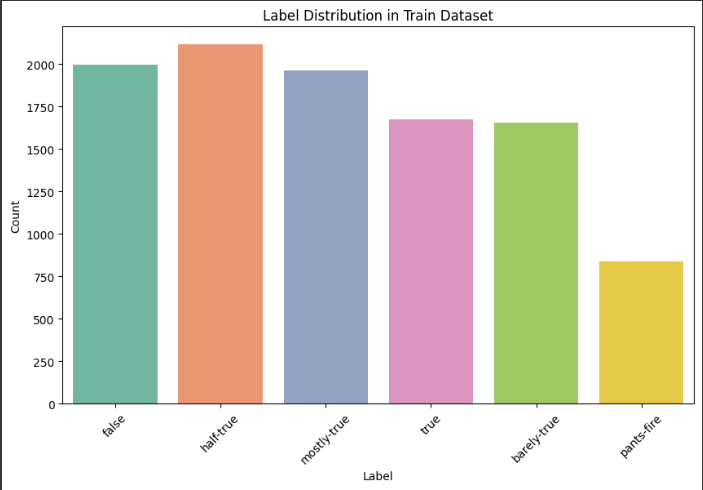


Figure 2: before SMOTE

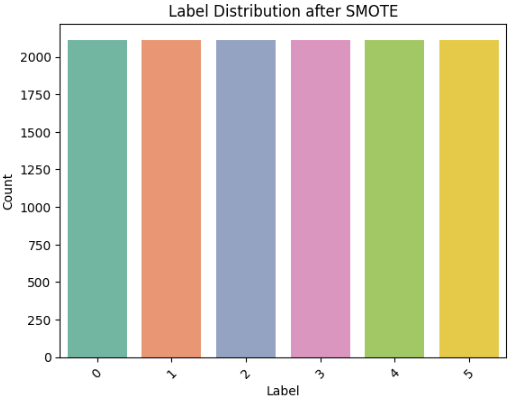


Figure 3: After SMOTE

1. *Evaluation Metrics*

For evaluation of the model, we used accuracy, precision, recall, F1-score, and confusion matrix.

**Results**

1. The distribution of sentiment scores is heavily centered on the neutral mark, indicating that most political statements in the dataset are phrased in a relatively neutral manner, with fewer statements showing extreme positivity or negativity. We observed that there is no correlation between the label of the dataset and the sentiment score. The sentiment score just indicates the tone of the statement.

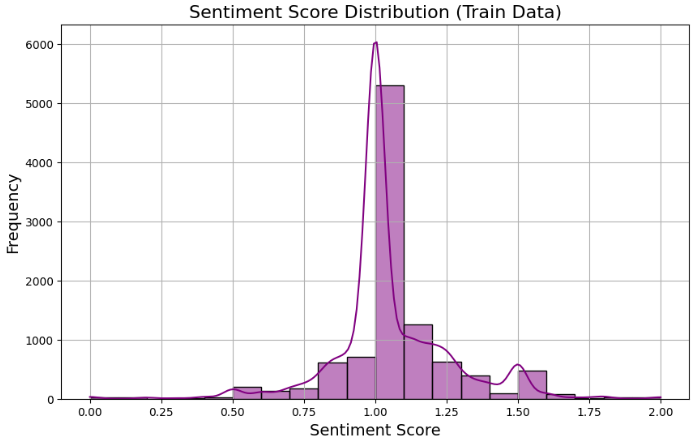


Figure 4

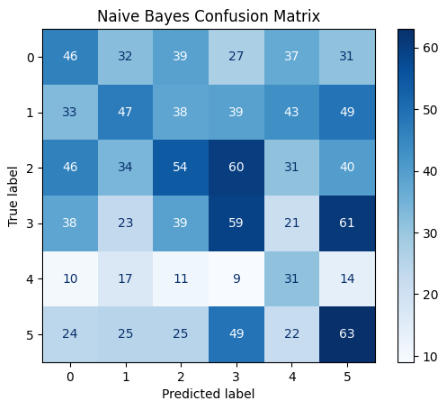
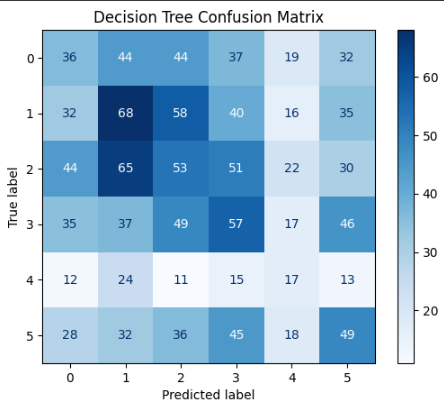
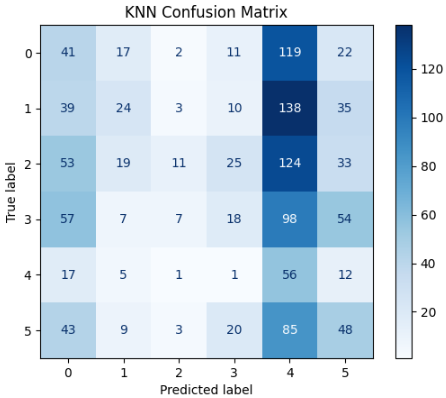
1. Naive Bayes showed an accuracy of approximately 23.68%. Figure 5 shows the confusion matrix of the Naive Bayes model. The square where True Label is equal to the Predicted Label is diagonal and dark it means, correct predictions (high true positives). Off-diagonal dark squares refers to lots of misclassifications (the model often confuses these two labels). *The confusion matrix illustrates the distribution of model predictions versus true labels. Darker squares along the diagonal indicate correct predictions with higher frequency, while darker off-diagonal squares highlight frequent misclassifications between similar classes.*  
   

Figure 5

1. Decision tree showed an accuracy of approximately 22.10% and it seems to overfit generally. In figure 6 you can observe the confusion matrix for Decision Tree  
     
    Figure 6
2. K-Nearest Neighbors model with 5 neighbors, exhibited an accuracy of approximately 16%. It performed worse than both Naive Bayes and Decision tree. It worked slowly too. Figure 7 shows the confusion matrix of KNN model  
     
    Figure 7
3. Logistic Regression showed an accuracy of 24.15% which is slightly better than Naive Bayes but almost similar. Logistic Regression outperforms Naive Bayes because unlike Naive Bayes, which assumes independence between features, Logistic Regression accounts for possible correlations between features. This makes it more robust when dealing with the complexities in the LIAR dataset. Refer to figure 8 for the visual representation of the confusion matrix.

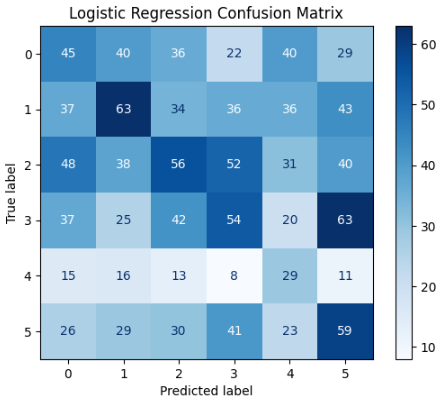
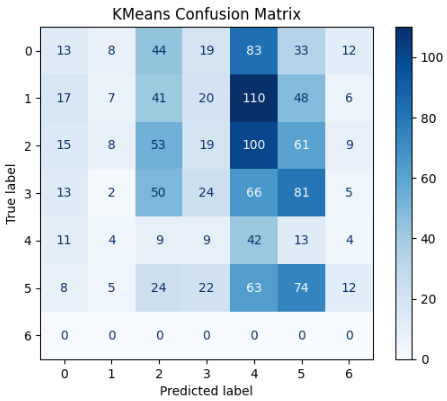


Figure 8

1. We explored the unsupervised algorithm of clustering too. We chose 7 clusters and checked the homogeneity score which was 0.0183. This indicated that the clustering doesn’t separate the data well. We tried to reduce the number of clusters or increase it but 7 seems to be the most optimal number of clusters. The result is as low as this because text data is tricky to cluster since the context and meaning behind the words are not captured well by just TF-IDF alone. Refer to figure 9 for the confusion matrix.   
     
     
    Figure 9
2. Figure 10 shows the comparison of model performance scores for Naive Bayes, Decision Tree, Logistic Regression, K-Nearest Neighbors (KNN) with 5 neighbors, and KMeans Clustering. The scores represent classification accuracy for supervised models and homogeneity score for the unsupervised KMeans model. Naive Bayes achieved the highest accuracy among the models evaluated.
3. To demonstrate the effectiveness of our model, we developed a Streamlit-based web application. The app allows users to input news headlines and instantly receive a prediction of whether the news is real or fake. Along with this, the app also provides a sentiment analysis of the input text, helping users understand the emotional tone of the statement. The Logistic Regression model, which showed the highest accuracy in the evaluation metrics, was deployed in the app. The app provides a user-friendly interface for interacting with the model, showcasing the practical use of fake news detection in real-time.

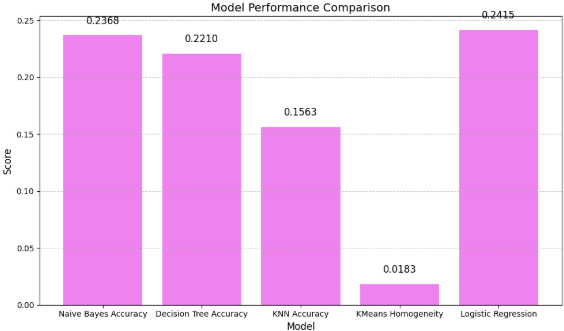


Figure 10

**Conclusion**

This project explored the application of machine learning algorithms for classifying textual statements based on their truthfulness. Several algorithms were implemented and evaluated, including Naive Bayes, Decision Tree, and K-Nearest Neighbors (KNN), along with the unsupervised K-Means Clustering. To address class imbalance and enhance model performance, oversampling techniques were applied to balance the dataset.

Among all the models tested, Logistic Regression achieved the highest accuracy. This can be attributed to its ability to perform better with non-linear feature relationships, as it uses the logistic function to map predictions to probabilities. In contrast, Naive Bayes, while simple and efficient, struggled with its assumption of feature independence, leading to lower performance. Despite these attempts, the overall performance remained low due to the inherent complexity and subtleties of natural language data.

To deepen the analysis, sentiment analysis was incorporated to assess the tone of the statements, providing additional insights. However, sentiment analysis only led to modest gains, suggesting that emotional tone alone is insufficient for accurate fake news detection.

There were several limitations in the study:

1. Simpler models struggled to capture the complex linguistic patterns inherent in the data.
2. While sentiment analysis contributed some improvement, it was not a major factor in boosting accuracy, highlighting the challenges in detecting fake news based on tone alone.
3. Despite the use of oversampling techniques, some minority classes remained challenging to classify.

The consistent low accuracy (~24%) across both Naive Bayes and Logistic Regression suggests that the issue may lie not with the model selection but with the quality of the underlying dataset, preprocessing techniques, or feature representation.

For future work, improvements can be made in the preprocessing stage, with a particular focus on handling class imbalance more effectively. Additionally, exploring deep learning approaches like LSTMs or transformer-based models (such as BERT) could significantly enhance performance. Transformer models, ensemble learning techniques, and even sarcasm detection modules could also be integrated to further improve fake news detection capabilities.

**References**

[1] *W. Y. Wang, "Liar, Liar Pants on Fire: A New Benchmark Dataset for Fake News Detection," Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017.*